IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.: 10/724,960	**Conf. No.:** 3575
Applicant: Biran	**TC/AU:** 2616
Filed: 12/01/2003	**Examiner**: Wong, Xavier S.
Customer No.: 45094	**Docket**: FIS920030298US1 (IBMF-0045)

Title: PARALLEL TCP SENDER IMPLEMENTATION

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

BRIEF OF APPELLANT

This is an appeal from the Final Rejection (Office Action) dated September 18, 2007, rejecting claims 1-3 and 5-20. The requisite fee set forth in 37 C.F.R. §1.17 (c) was submitted on December 18, 2007.

REAL PARTY IN INTEREST

International Business Machines Corporation is the real party in interest.

RELATED APPEALS AND INTERFERENCES

There is no related appeal or interference.

STATUS OF CLAIMS

As filed, this case included claims 1-20. Claims 1-3 and 5-20 remain pending, stand rejected, and form the basis of this appeal, claim 4 has been cancelled. No claim has been allowed. The rejections of claims 1-3 and 5-20 are being appealed.

STATUS OF AMENDMENTS

No after-final amendment of claims was proposed following the Final Rejection of September 18, 2007.

SUMMARY OF THE CLAIMED SUBJECT MATTER

Independent claim 1 provides a transfer control protocol (TCP) transmission system (10), comprising: a transmit request handler (12) that receives request events (23) from a requester (22) including a notification of new transmission data posted (38, para. 0019), records event information into a connection context (para. 0014) and, based on the notification of new transmission data posted, either schedules a connection in a ready queue (18) or places the connection in a pending queue (16) (para. 0019); and a transmitter (14) that operates in parallel with the transmit request handler (12) (para. 0013), wherein the transmitter (14) dequeues connections from the ready queue and prepares packets for transmission based on information recorded in the connection context (para. 0013).

Independent claim 10 provides a method for transmitting packets in a transfer control protocol (TCP) environment (10), comprising: receiving a request event at a

transmit request handler, the request event including a notification of new transmission data posted (para. 0019); processing the request event in the transmit request handler (12) to, based on the notification of new transmission data posted, either schedule a connection in a ready queue (18) or place the connection in a pending queue (16) (para. 0019); providing a transmitter (14) that operates in parallel with the transmit request handler (para. 0013); and utilizing the transmitter to dequeue connections from the ready queue and prepare packets for transmission (para. 0013).

Independent claim 15 provides a system for transmitting packets in a transfer control protocol (TCP) environment, comprising: a connection context (20) for storing event information (para. 0014); a transmit request handler (12) that receives request events from a requester including a notification of new transmission data posted (para. 0019), records the event information into the connection context (20) and, based on the notification of new transmission data posted, either schedules a connection in a ready queue (18) or places the connection in a pending queue (16) (para. 0019); a transmitter (14) that operates in parallel with the transmit request handler (para. 0013), wherein the transmitter dequeues connections from the ready queue and prepares packets for transmission based on event information stored in the connection context (para. 0019); and a queue manager (28) for moving connections from the pending queue to the ready queue (para. 0017).

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Whether claims 1-3, 5-6, 8-10 and 12-20 are unpatentable under 35 U.S.C. §103(a) over Jayam et al. (US Pub 2003/0115337 A1), hereinafter "Jayam", in view of Kagan et al. (US Pub 2002/0152315 A1), hereinafter "Kagan".

2. Whether claims 7, 11 and 18 are unpatentable under 35 U.S.C. §103(a) over Jayam and Kagan and further in view of Buskens et al. (US Patent 5905871), hereinafter "Buskens".

ARGUMENTS

1. Claims 1-3, 5-6, 8-10 and 12-20 are not obvious under 35 U.S.C. §103(a) over Jayam in view of Kagan.

Appellants submit that Jayam and Kagan do not teach a transmit request handler and a transmitter operating in parallel as claimed in the current application. Specifically, the claimed transmit request handler upon receiving request events either schedules a connection (not a packet) in a ready queue or places the connection in a pending queue, and in parallel, a transmitter dequeues connections from the ready queue and prepares packets for transmission based on respective connection contexts. (Claim 1, similarly claimed in claims 10 and 15.) Jayam does not disclose or suggest such parallel operations between two different components, i.e., transmit request handler and transmitter. In Jayam, the transmit control block (TxTCB) processes the to-be-transmitted information including a packet or a pointer to the packet. (Para. 0046). As a TxTCB of Jayam already has the packet or pointer to the packet, it further determines

whether the transmit window is sufficient to transmit the packet. (Para. 0047). In Jayam, it is the packet that is either put into a transmit pending queue or transmitted and put into the retransmit queue. (Id.) This is clearly different than the claimed invention because Jayam does not generate connections (before packets) and does not put the connects into different queues. Jayam also does not include a separate transmitter that generates packets from the connection contexts of the connections in the ready queue. Jayam does not disclose or suggest parallel operations by two different components for generating connections based on received request events and preparing packets from connection contexts of connections in a ready queue, respectively.

In addition, Jayam does not include a ready queue and a pending queue that both contain connections, not packets. Note that a connection does not include data or pointer to data ready to be transmitted. (*See* current specification at para. 00018, "no data is available yet.") A connection context "is essentially a data structure that stores data pertaining to the particular connection." (Id. at para. 00014). On the contrary, a packet includes data segments ready to be transmitted and is generated from the respective connection context. In Jayam, either the transmit pending queue or the retransmit queue (which contains copies of the transmitted packets or their pointers) contains packets or their pointers, not connections as in the claimed invention. (Para. 0047 of Jayam).

The Examiner also asserts that "the processes of TxTCB is run in parallel with the host computer-transmission destination (paragraph 0059); therefore, the request handling and transmission are run in parallel." (Office Action at page 4). Appellants

respectfully disagree because in Jayam a TxTCB is part of its respective host such that it does not operation in parallel to the host. Further, it is irrelevant whether a transmission destination host runs in parallel to the transmitting host or the TxTCB thereof because the claimed invention claims the parallel operations in receiving/processing requests to generate connections and preparing packet from connection contexts for transmission, not in transmitting data and receiving the same data. Furthermore, in Jayam, the host computer does not operate in parallel to the TxTCB to dequeue connections in a ready queue and prepare packets from the respective connection contexts. The Examiner does not even assert how the claimed feature of preparing packets from connection contexts is suggested by Jayam and/or Kagan.

Kagan does not overcome the above deficiencies of Jayam.

The Examiner relies on Kagan only to overcome the admitted deficiency of Jayam with respect to the claimed limitation "...based on the notification of new transmission data posted, either schedules a connection in a ready queue or places the connection in a pending queue[.]" (Claim 1, similarly claimed in claims 10 and 15). However, Kagan also does not teach this feature. In Kagan, the descriptor 38 is written to indicate the source of the data to be sent in memory 32 and the destination thereof. (*See* para. 0046.) Kagan does not disclose or suggest that descriptor 38 indicates that the data transmission is "new" such that descriptor 38 does not function as a notification of new transmission data posted. Note that Kagan does not use the descriptor 38 to indicate whether a previously incomplete connection is completed by the new transmission data. The Examiner asserts that "the description of 'rings a doorbell' is

obvious that data/descriptor requires some sort of acknowledgement due to 'unfamiliarity of data coming in,' and therefore, [is] considered as 'new' data, before the data is allowed to enter the HCA." (Advisory Action at page 2). Appellants submit that that this interpretation clearly distorts Kagan as Kagan only uses the doorbell as an interface between host 24 and HCA 22, for notifying the HCA that there are descriptors 38 awaiting execution on a particularly QP. (Para. 0046). Kagan does not imply or suggest, and it is not necessary, that the descriptor 38 needs to be new. The Examiner apparently confuses the "doorbell" operation in Kagan with the doorbell of a physical world house in making the statement of "unfamiliarity of data coming in", which is inconsistent with the disclosure of Kagan. Appellants submit that even in the physical world, a person ringing a doorbell of a house does not necessarily indicate that the person is unfamiliar to the household.

In view of the foregoing, Appellants submit that Jayam and Kagan, even in the suggested combination, do not disclose or suggest each and every claimed limitation. The dependent claims are believed allowable for the same reasons as well as for their own additional features.

In view of the foregoing, Appellants respectfully request reversal of the final rejection.

2. Claims 7, 11 and 18 are not obvious under 35 U.S.C. §103(a) over Jayam and Kagan and further in view of Buskens.

The above arguments also applied to claims 7, 11 and 18 as Buskens does not overcome the identified deficiencies of Jayam and Kagan. In view of the foregoing, Appellants respectfully request reversal of the final rejection.

In view of the foregoing, Appellants submit that the final rejection is defective, and should be reversed.

Respectfully submitted,

/Jianping Zhang/

Jianping Zhang
Reg. No. L414

Dated: February 19, 2008

Hoffman, Warnick & D'Alessandro LLC
75 State Street 14th Floor
Albany, NY 12207
Telephone: (518) 449-0044
Fax: (518) 449-0047

CLAIMS APPENDIX

1. A transfer control protocol (TCP) transmission system, comprising:

a transmit request handler that receives request events from a requester including a notification of new transmission data posted, records event information into a connection context and, based on the notification of new transmission data posted, either schedules a connection in a ready queue or places the connection in a pending queue; and

a transmitter that operates in parallel with the transmit request handler, wherein the transmitter dequeues connections from the ready queue and prepares packets for transmission based on information recorded in the connection context.

2. The TCP transmission system of claim 1, wherein the ready queue comprises a linked list of connections.

3. The TCP transmission system of claim 1, wherein each connection comprises a connection context data structure.

5. The TCP transmission system of claim 1, wherein the request events include a request to send an acknowledgement.

6. The TCP transmission system of claim 1, wherein the request events include a request for a window update.

7. The TCP transmission system of claim 1, wherein the request events include handling of an incoming acknowledgement.

8. The TCP transmission system of claim 1, further comprising a queue manager, wherein the queue manager includes means for receiving a timer expiration.

9. The TCP transmission system of claim 1, wherein the transmitter includes:

means for deciding what type of segment should be transmitted;

means for building a transmit command and requesting segment data; and

means for building header information for the segment being transmitted.

10. A method for transmitting packets in a transfer control protocol (TCP) environment, comprising:

receiving a request event at a transmit request handler, the request event including a notification of new transmission data posted;

processing the request event in the transmit request handler to, based on the notification of new transmission data posted, either schedule a connection in a ready queue or place the connection in a pending queue;

providing a transmitter that operates in parallel with the transmit request handler; and

utilizing the transmitter to dequeue connections from the ready queue and prepare packets for transmission.

11. The method of claim 10, wherein the request event is further selected from the group consisting of: a request to send an acknowledgement, a request for a window update, and a request to handle an incoming acknowledgement.

12. The method of claim 10, further comprising moving connections from the pending queue to the ready queue.

13. The method of claim 10, wherein the dequeuing connections from the ready queue and prepare packets for transmission, includes:

handling any timeouts of a retransmit timer;

deciding on a type of segment to transmit;

building a transmit command and requesting segment data;

building headers;

recording information on a previously transmitted segment; and

starting a retransmit timer if data was transmitted.

14. The method of claim 13, wherein each connection includes data describing the request event in a connection context.

15. A system for transmitting packets in a transfer control protocol (TCP) environment, comprising:

a connection context for storing event information;

a transmit request handler that receives request events from a requester including a notification of new transmission data posted, records the event information into the connection context and, based on the notification of new transmission data posted, either schedules a connection in a ready queue or places the connection in a pending queue;

a transmitter that operates in parallel with the transmit request handler, wherein the transmitter dequeues connections from the ready queue and prepares packets for transmission based on event information stored in the connection context; and

a queue manager for moving connections from the pending queue to the ready queue.

16. The system of claim 15, wherein the ready queue comprises a linked list of connections.

17. The system of claim 15, wherein each connection comprises a connection context data structure.

18. The system of claim 15, wherein the request event is selected from the group consisting of: a notification of new transmission data posted, a request to send an acknowledgement, a request for a window update, and a request to handle an incoming acknowledgement.

19. The system of claim 15, wherein the transmitter includes:

means for deciding what type of segment should be transmitted;

means for building a transmit command and requesting segment data; and

means for building header information for the segment being transmitted.

20. The system of claim 15, wherein the queue manager receives any timer expirations.

EVIDENCE APPENDIX

There is no evidence submitted.

RELATED PROCEEDINGS APPENDIX

There is no related proceeding.

CERTIFICATE OF SERVICES

There is no other party to this appeal proceeding.